L Number	Hits	Search Text	DB	Time stamp
1	2	(litigation adj support) and (image adj files!) and ((convert$3 or conversion) near9 image)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/15 10:00
2	2157	(image adj files!) and ((convert$3 or conversion) near9 image)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/15 09:35
3	561	((image adj files!) and ((convert$3 or conversion) near9 image)) and (different adj2 (formats! or types! or files!))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/15 10:01
4	3	(((image adj files!) and ((convert$3 or conversion) near9 image)) and (different adj2 (formats! or types! or files!))) and (export or exported or exporting) near3 (image adj file)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/15 09:37
5	4	(((image adj files!) and ((convert$3 or conversion) near9 image)) and (different adj2 (formats! or types! or files!))) and (export or exported or exporting) near9 (image adj file)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/15 09:37
6	418	(((image adj files!) and ((convert$3 or conversion) near9 image)) and (different adj2 (formats! or types! or files!))) and log$6	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/15 09:38
7	125	(((image adj files!) and ((convert$3 or conversion) near9 image)) and (different adj2 (formats! or types! or files!))) and (log or logged or logging) and (classif$6 or categori$6 or organiz$6)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/15 09:48
8	3	((((image adj files!) and ((convert$3 or conversion) near9 image)) and (different adj2 (formats! or types! or files!))) and (log or logged or logging) and (classif$6 or categori$6 or organiz$6)) and (SHA! or (secure adj hash adj algorithm) or hash$6) with (computing or compute or computed or computing or calculat$6)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/15 09:41
9	3	(((image adj files!) and ((convert$3 or conversion) near9 image)) and (different adj2 (formats! or types! or files!))) and (SHA! or (secure adj hash adj algorithm) or hash$6) with (computing or compute or computed or computing or calculat$6)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/15 09:41
10	14	((image adj files!) and ((convert$3 or conversion) near9 image)) and (SHA! or (secure adj hash adj algorithm) or hash$6) with (computing or compute or computed or computing or calculat$6)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/15 09:42
11	3	((image adj files!) and ((convert$3 or conversion) near9 image)) and (SHA! or (secure adj hash adj algorithm) or hash$6) with (computing or compute or computed or computing or calculat$6) with duplicat$4	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/15 09:43
12	18	(SHA! or (secure adj hash adj algorithm) or hash$6) with (computing or compute or computed or computing or calculat$6) with duplicat$4	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/15 09:43
13	44	((((image adj files!) and ((convert$3 or conversion) near9 image)) and (different adj2 (formats! or types! or files!))) and (log or logged or logging) and (classif$6 or categori$6 or organiz$6)) and automat$6 and ((electronic adj (mail or document or messag$3)) or email or e-mail)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/15 10:01

14	2051	(image adj files!) and ((convert$3 or conversion) near6 image)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/15 10:01
15	222	((image adj files!) and ((convert$3 or conversion) near6 image)) and (different adj2 (formats! or files!))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/15 10:01
16	32	(((image adj files!) and ((convert$3 or conversion) near6 image)) and (different adj2 (formats! or files!))) and (automat$6 near2 (system or manag$6)) and ((electronic adj (mail or document or messag$3)) or email or e-mail)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/15 10:03
17	10	((((image adj files!) and ((convert$3 or conversion) near6 image)) and (different adj2 (formats! or files!))) and (automat$6 near2 (system or manag$6)) and ((electronic adj (mail or document or messag$3)) or email or e-mail)) and (707/$.ccls. or 715/$.ccls.)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/15 10:11
18	25328	707/$.ccls. or 715/$.ccls.	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/03/15 10:11

Search Results

Search Results for: **[(litigation <near/2> support)]**
Found **20** of **127,944 searched.**

Search within Results

(different <near/2> (type or format or file)) GO > Advanced Search

> Search Help/Tips

Sort by: Title Publication Publication Date Score Binder

Results 1 - 20 of 20 short listing

1 Conceptual information retrieval in litigation support systems — 100%
Vijay Mital , Agathoclis Stylianou , Les Johnson
Proceedings of the third international conference on Artificial intelligence and law May 1991

2 Beyond boolean search: FLEXICON, a legal tex-based intelligent system — 100%
Dephne Gelbart , J. C. Smith
Proceedings of the third international conference on Artificial intelligence and law May 1991

3 An example of integrating legal case based reasoning with object-oriented rule-based systems: IKBALS II — 100%
George Vossos , John Zeleznikow , Tharam Dillon , Vivian Vossos
Proceedings of the third international conference on Artificial intelligence and law May 1991

4 Optical disks (panel session): effecting successful integration — 100%
D'Ellen Bardes , Patrick Call , Michael S. Theis , Taroon C. Kamdar
Proceedings of the 1985 ACM annual conference on The range of computing : mid-80's perspective: mid-80's perspective October 1985

5 "Computers and the law": A course description — 100%
Philip Koltun
ACM SIGCSE Bulletin , Proceedings of the tenth SIGCSE technical symposium on Computer science education January 1979
Volume 11 Issue 1
This paper surveys current legal issues in the computer field and discusses the author's experience in twice teaching a course on computers and the law at

Sangamon State University. Legal topics highlighted in the course include databanks and privacy; contracting for computer services; protection of proprietary software; tax treatment of computer software; liability problems in providing computer services; government regulation of the computer and communications industries; and the Justice ...

6 The design of an attorney's statistical consultant 100%
L. S. Lutomski
Proceedings of the second international conference on Artificial intelligence and law May 1989

7 Virtual artefacts to support negotiation within an augmented collaborative environment for alternate dispute resolution 100%
Peter Dew , Aphrodite Galata , John Maxfield , Daniela Romano
Proceedings of the 4th international conference on Collaborative virtual environments September 2002

Commercial organisations that are in dispute will sometimes seek to settle their differences without resorting to legal proceedings through a process of structured and facilitated negotiation called Alternate Dispute Resolution (ADR). Traditionally ADR negotiations are conducted by co-locating the mediator, witnesses, supporting evidence and representatives of each party. This paper describes a novel experiment that has been conducted at the University of Leeds to investigate whether an augmente ...

8 A connectionist approach to conceptual information retrieval 100%
R. K. Belew
Proceedings of the first international conference on Artificial intelligence and law December 1987

This report proposes that recent advances using low-level connectionist representations offer new possibilities to those interested in free text information retrieval (IR). The AIR system demonstrates that this representation suits the IR domain well, particularly the special problems attending the more sophisticated forms of conceptual retrieval required in legal applications. Also, the natural way in which connectionist representations allow learning means that AIR can av ...

9 Focus issue on legacy information systems and business process engineering: the role of stakeholders in managing change 100%
Elayne Coakes , Tony Elliman
Communications of the AIS July 1999

10 Social responsibility and the CS student: how can I get involved? 100%
Saul Jimenez
Crossroads May 1995
Volume 1 Issue 4

11 An evaluation of retrieval effectiveness for a full-text document-retrieval system 100%
David C. Blair , M. E. Maron
Communications of the ACM March 1985
Volume 28 Issue 3

An evaluation of a large, operational full-text document-retrieval system (containing

roughly 350,000 pages of text) shows the system to be retrieving less than 20 percent of the documents relevant to a particular search. The findings are discussed in terms of the theory and practice of full-text document retrieval.

12 An application of rule-based and case-based reasoning within a single legal knowledge-based system 100%
Kamalendu Pal , John A. Campbell
ACM SIGMIS Database September 1997
Volume 28 Issue 4

A knowledge-based system, Advisory Support for Home Settlement in Divorce (ASHSD), which gives advice on different aspects of matrimonial-home-settlement in English divorce law is described. The system employs two reasoning methods, rule-based reasoning (RBR) and case-based reasoning (CBR) in an integrated framework. It automates the estimation of the relative suitability of these reasoning methods for any given new case. This relative suitability is judged by matching the features of the select ...

13 Toward knowledge management systems in the legal domain 100%
Deborah L. Edwards , Dirk E. Mahling
Proceedings of the international ACM SIGGROUP conference on Supporting group work : the integration challenge: the integration challenge November 1997

14 More than wyshful thinking: AustLII's legal inferencing via the World Wide Web 100%
Graham Greenleaf , Andrew Mowbray , Geoffrey King , Simon Cant , Philip Chung
Proceedings of the sixth international conference on Artificial intelligence and law June 1997

15 Animation and imaging for litigation support 100%
Dennis M. Brewer , Charlton B. Ogden
ACM SIGGRAPH 96 Visual Proceedings: The art and interdisciplinary programs of SIGGRAPH '96 January 1996

16 Making work visible 100%
Lucy Suchman
Communications of the ACM September 1995
Volume 38 Issue 9

17 Information filtering: the computation of similarities in large corpora of legal texts 100%
Erich Schweighofer , Werner Winiwarter , Dieter Merkl
Proceedings of the fifth international conference on Artificial intelligence and law May 1995

18 Hypertext for the electronic library?: CORE sample results 100%
Dennis E. Egan , Michael E. Lesk , R. Daniel Ketchum , Carol C. Lochbaum , Joel R. Remde , Michael Littman , Thomas K. Landauer
Proceedings of the third annual ACM conference on Hypertext September 1991

19 Stop the Presses: Ownership of Linux Trademark Resolved 99%
Margie Richardson
Linux Journal November 1997

20 What makes professionals so difficult: an investigation into professional ethics teaching 99%
David Preston
ACM SIGCAS Computers and Society , Proceedings of the ethics and social impact component on Shaping policy in the information age June 1998
Volume 28 Issue 2

Teaching ethics to professionals pursuing a university degree programme requires a method that engages them with the realities and problematic nature of their workplace environment. In this paper we examine some of the history of Professional Ethics from a philosophical and political standpoint. Unfortunately this analysis appears to produce more questions than answers with the terms professional and expert seemingly poorly defined. In order to demonstrate some of the generic problems likely to ...

Results 1 - 20 of 20 **short listing**

CONCEPTUAL INFORMATION RETRIEVAL IN LITIGATION SUPPORT SYSTEMS

Vijay Mital, Agathoclis Stylianou and Les Johnson
Knowledge Based Systems Group
Computer Science Department
Brunel University
Uxbridge, Middlesex UB8 3PU, U.K.
+44 0895 74000; se89vvm@cs.brunel.ac.uk

Abstract

This paper is concerned with information retrieval in the context of supporting complex litigation by managing large numbers of documents. It is shown that the application is sufficiently different from searching for case/statute text or reasoning with the law, so as to render the techniques developed for the latter inappropriate. A new approach to information representation and system design is identified and developed. The paper presents an architecture that takes into account the peculiar characteristics of the application and enables the utilisation of existing skills of professionals, thereby facilitating rapid and consistent encoding. An extended object-oriented paradigm underlies the architecture. Using this paradigm, it has been possible to combine techniques developed for large databases with the purposive or functional similarity approach to search and retrieval taken in case-based design systems.

1 Introduction

The problems associated with full or free text retrieval are well known. Even where thesauri (Bing, 1989) and lexicons (Weaver et al, 1989) are employed, users find it difficult to formulate queries to pinpoint those out of a large collection of documents that might contain the desired information. It is possible to improve the user interface, e.g., by means of a front-end containing rules associating concepts of interest with particular word/phrase patterns (Lewis et al, 1989; Tong et al, 1989). However, the fact remains that, in a rich domain, words and phrases are a poor approximation to meaning without due consideration of the conceptual relationships between them (Rau, 1987).

Researchers concerned with artificial intelligence applications in law have confronted the above issues directly because information retrieval is a task integral to most such applications. They have long recognised the need to organise legal information in a manner that enables retrieval based on the meaning and legal significance of text (Hafner, 1981; Bertaina et al, 1982). The purpose of retrieval may vary. A legal research system may simply display the located information (Dick, 1987); a case-based reasoning system may itself make the inferences (Rissland & Ashley, 1989); or the position may be somewhere in between (Gelbart & Smith, 1990). The common element is the aim to represent the relationships and dependencies between a legal concept and its subconcepts, and between a concept and its categorisation in the universe of discourse (Bareiss, 1989), in as explicit a manner as the technology permits[1]. As either complex generalised formalisms such as conceptual dependencies (Schank, 1975) or sophisticated special-purpose representations (Cross & deBessonet, 1985) are employed to make these relationships and dependencies explicit, theoretically, any (and every) element of legal significance can be indexed upon and differentiated from any other element.

However, the process of indexing on or reasoning with complex knowledge structures is computationally expensive, to the point of being intractable when, say, more than a few hundred realistic documents or cases are to be dealt with (Martin, 1989). It is not being suggested that the problems will not be overcome in due course if research continues apace along the present lines and significant advances continue to be made. But, there is another factor which makes some aspects of the

above approach inappropriate for the purposes with which we are concerned in this paper. There are some profound dissimilarities between the computerised support of complex litigation by management and retrieval of documents on one hand and legal research or legal reasoning with a view to giving advice on the other.

It is true that the concepts that a lawyer is interested in when using a litigation support system (LSS) are essentially the same as would be recorded in the judgement of the court. The decoupling of the text actually used and the concepts it conveys has the same effect. For instance, a search of a database of reported cases may reveal that the word 'intention' does not appear in an exposition of the concept of intention (Dick, 1987). Whilst, a user of a LSS may find that letters and memoranda discussing an 'accident' contain only oblique references such as 'what happened last week' and '[w]e all know why we're here' (Blair & Maron, 1985). Or, that the reference in a letter to a 'defect' is in graphic terms - 'our product explodes' - but the word 'defect' is absent (Wallwork, 1989). It is at the practical level - and practical considerations exert great influence on system design (Mital & Johnson, 1991a) - that the litigation support application most significantly differs. Some of the special application characteristics which have influenced the system architecture presented in this paper are as follows:

(a) A litigation support database often has to endure large, spasmodic, additions[2].
(b) Usually, every document to be inserted in the database will have been read and screened for relevance by one or more members of the litigation team - either junior lawyers, or paralegals. These persons have the ability to abstract information from documents for the purpose of indexing/cataloguing documents in a relatively consistent manner, either manually (Halverson, 1979) or using one of the widely available LSS that are loosely based on the manual indexing/cataloguing methodology but do not purport to effect conceptual retrieval (Wilkins, 1989; Christian, 1990).
(c) Any particular LSS is likely to be used only by a small number of persons whose profile is predictable in advance, as is the role of the system.

Consequently, the representation formalism should allow rapid encoding by a number of persons working without constant reference to each other. The aspect of information that is to be abstracted and represented should be such as the personnel already in place are capable of providing. Lastly, it makes sense to sacrifice generality of representation - even assuming that it is achievable - in favour of giving the users the means to adapt the representation schema to their own peculiar needs. The consideration of computational tractability, so that large numbers of documents may be handled, has already been mentioned.

In subsequent sections, we present the architecture of a system currently under development by the authors that is based on an object-oriented schema which we believe to be particularly suitable for the information characteristics at hand. This system is not intended to *replace* full text retrieval systems, but to *augment* them. We combine techniques developed for object-oriented databases that can handle vast quantities of simply represented information, with the rich notion of retrieval on the basis of purposive or functional similarity usually employed in case-based design (CBD) systems. We start by briefly mentioning the characteristics of the object-oriented paradigm as extended for the application at hand.

2 Extended Object-Oriented Paradigm

The object-oriented (OO) paradigm is based on the idea of abstracting the characteristics of a world truth in a manner that has a direct and natural correspondence between the world and its model, and encapsulating that abstraction. Objects contain a data structure and, in addition, may contain the procedures (methods) associated with the data. There is as yet no established form for the OO paradigm, with application specific adaptations prevailing. Broadly speaking, it is possible to distinguish those applications where the task involves systems analysis or program construction from those where the emphasis is on the richness of representation of data (Kim, 1990) or knowledge (Patel-Schneider, 1990). In the latter manifestations, the OO paradigm formalises and extends the representational ideas underlying semantic networks and frames.

Objects can be classified in a taxonomy formed by generalisation-specialisation or parent-child links. If multiple parents are permitted, the taxonomy may be termed a tangled hierarchy. The primary functions of the parent-child links are to enable properties or information to be inherited and to allow limited inferences to be made. These links are not meant to be determinative of the semantics of the relationship between two concepts that happen to be placed as a parent and a child or siblings in a taxonomy for some limited purpose. Other kinds of links are, therefore, added to the core formulation to represent the required information explicitly and elegantly.

One such link is the association link which can be used to relate two object classes. If the link is treated as an object in its own right, then an instance of the link connects an instance of each of the related classes; the semantics of different kinds of association links can be carried as data and methods within the link, rather than in each of the linked classes.

Another kind of link, which has the same surface structure as an association link, is also sometimes necessary. An example is given in Figure 1, where it is sought to represent the information that an instance of class X can never co-exist with any instance of class Y. This, obviously, is not easy to represent using parent-child links. It is more naturally represented using a link labelled 'cannot exist together' between the two classes. The 'cannot exist together' link in the example in Figure 1 should not be confused with an association link because the latter is constrained into showing the functional dependency and connectivity between the associated objects. Therefore, the link shown in Figure 1 is termed an *extended association link.*

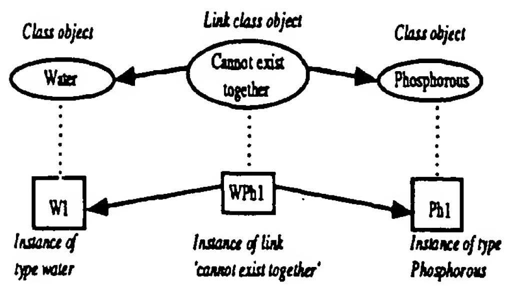

Figure 1 : Conceptual Structure of Extended Association Link

3 Representing the Purpose, Not Entire Text Content

The constructs which we provide do not readily allow *all* the information contained in documents to be represented. In fact, we have no desire to have all the available information encoded, for that would lead to the same problems of retrieval and reasoning being overwhelmed with a surfeit of information that affect full text retrieval. If the entire contents are not to be represented, and only salient features are to be captured, what is salient must be clarified.

We start with the assumption that the user is primarily interested in the retrieval of documents for a particular purpose, and only those features that are relevant to that purpose need be explicit. The primary purpose of the user is, with the aid of documents, to prove or disprove certain legal or factual issues that are in contention - the reference is not just to the issues that a court might frame, but also to those that might be used for indexing/cataloguing documents either manually or using one of the commercially available LSS. A simple example can be given. In the case of an alleged negligent misrepresentation by a financial adviser leading to a loss being suffered by the client, the following may be some of the broad issues of contention, with further decomposition as shown in Figure 2:

(a) Whether the client possessed information from independent sources, such as to enable him to know that the defendant's advice was incorrect.

(b) Whether the client acted as per the defendant's advice.

(c) Whether the loss was caused by reasons other that the actions taken pursuant to the defendant's advice.

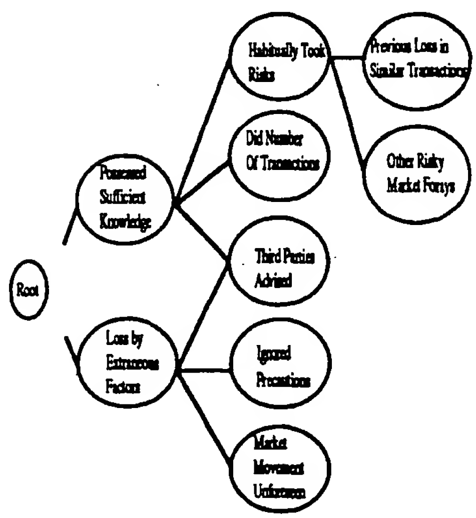

Figure 2 : Issues Taxonomy

The relevance of a retrieved document to one or more issues (or sub-issues) may be the ultimate test of how well it serves the purpose of the user. However, labelling each document merely with those issues to which it is thought to be relevant is too imprecise and coarse. This is because the user may wish to retrieve only those documents that relate to an issue *in a particular way.* For instance, documents that are relevant to the issue 'third parties advised', but are relevant by virtue of being admissions recorded by the client himself, rather than as statements in letters sent to the client by third party advisers, or records by other persons of conversations

between the client and the advisers. This is a rich notion of the purpose of retrieval and, as shall be discussed below, is akin to the notion of functionality or similarity in some CBD systems, including where design involves legal reasoning (Mital & Johnson, 1991b). In those systems, functionality of a case (or its similarity to the problem at hand) may be measured by the case's usefulness in solving a certain problem element in the context of the particular situation. This notion of functionality, when applied to litigation support, means that the relevance of a document to an issue is no longer merely a pre-defined and static parameter, but is judged dynamically depending upon the context specified by means of the request for retrieval. We seek to represent only that aspect of information which is required to serve such broad purposes of the user.

4 Conceptual Representation of Documents

The conceptual representation of a document is as an aggregate object containing instances of other objects (i.e., primitive domain concepts, issues, explanatory links, reference links and relevance functions), see Figure 3.

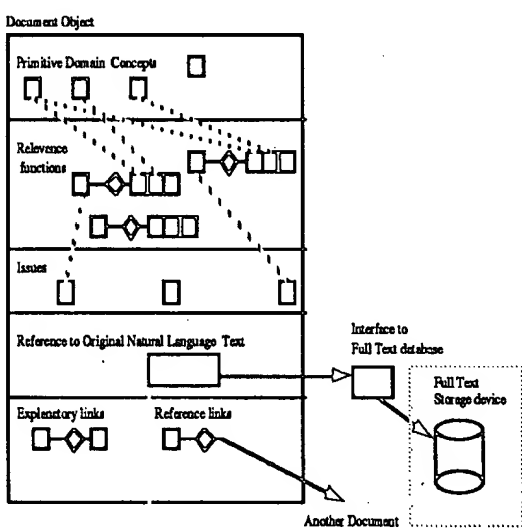

Figure 3 : Conceptual Representation of Document

The original natural language text is not held within the system being described here; it is anticipated that a full text database that relies on fast retrieval devices such as optical disks, perhaps with an industry standard query interface (Cornwell, 1990), will be employed. The facility for interfacing to SQL fronted relational databases has been provided (Stylianou, 1990).

The issues taxonomy has already been mentioned. Primitive domain concepts include (a) the basic facts which are used to judge the relevance of a document to an issue, and (b) the kinds of documents that occur in the domain of interest. They are organised in two separate tangled hierarchies, as illustrated in Figures 4(a) and 4(b).

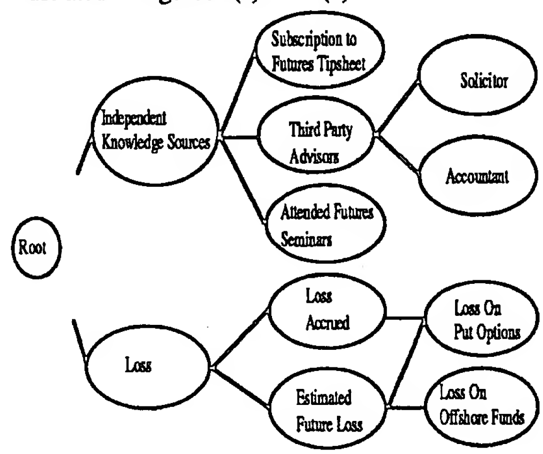

Figure 4(a) : Primitive Domain Concepts (Situation Facts)

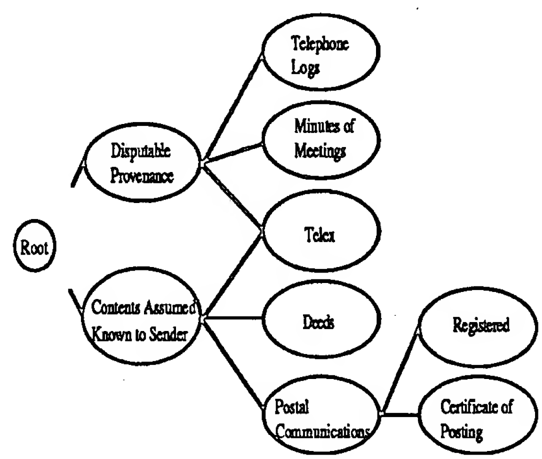

Figure 4(b) : Primitive Domain Concepts (Document-Kinds)

4.1 Explanatory Links

Explanatory links relate issues and primitive domain concepts in any one of the combinations shown in Figure 5, where PDC stands for primitive domain concept[3]. The links represent the extent of the validity of the particular interpretation regarding relevance that the person carrying out the encoding has placed on the document. For example, he may state that a document contains the primitive domain

concept "attended seminars on futures". However, there may be an alternative interpretation that the client did not attend the seminar as a passive listener and, instead, went to see the lecturer for personal advice on a particular problem. If so, the primitive domain concept 'advised by third parties' may be related to 'attended seminars on futures' using the 'alternative interpretation' link. It must be noted that explanations are confined to the document in which they are specified, and are not to be thought of as global relationships.

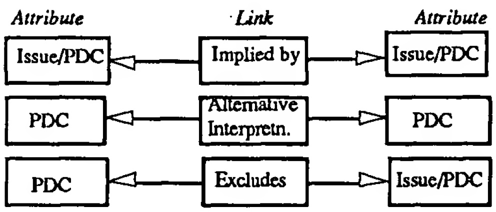

Figure 5 : Types of Explanatory Links

4.2 Reference Links

A reference link relates the document in which the link is specified to one or more other documents, see Figure 6. Each type of reference link has a special semantic significance. For example, a letter may be sent by the plaintiff in reply to an accusatory letter from the defendant, denying liability. If the encoder specifies the "rebuts" link, it signifies that the plaintiff's letter contains information likely to go to disprove some or all those issues in the proof of which the defendant's letter is relevant.

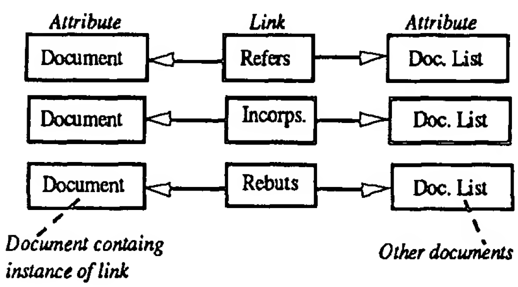

Figure 6 : Types of Reference Links

4.3 Relevance Functions

Relevance functions relate issues with primitive domain concepts. The attributes of a relevance function object, an instance of which is shown in Figure 7, are (a) one, and only one, issue; and (b) one or more primitive domain concepts.

Figure 7 : An Instance of the Relevance Function Link

A particular instance of a relevance function is, strictly speaking, valid only within the context of the document in which it is specified[4]. Still, it is inevitable that *instances* of relevance functions (RF's) containing identical attribute values will occur in a number of documents. Also, that different documents will contain RF's that have attribute values such as to make the documents 'similar' in some sense or for some purpose of the user. As documents are sought to be organised and indexed by the RF's they contain, the RF's themselves must be organised so that searching is minimised. We have chosen to use discrimination by hierarchical subsumption (Galloway, 1987) as the basis of organising RF's. The primary criterion for discrimination is the value of the issue (IU) attribute, the secondary criteria are the values of the primitive domain concept (PDC) attributes in the order of importance pre-specified by the human encoder. Figure 8 illustrates subsumption by discrimination[5].

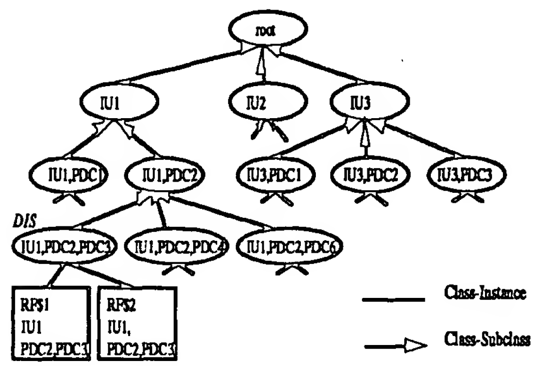

Figure 8 : Subsumption Hierarchy of RF's

Essentially, documents may be thought to be notionally organised *through* the RF subsumption hierarchy. However, as each document may contain more than one RF, the *documents themselves are not in a subsumption hierarchy.* This is in marked contrast to the simpler object or frame-based systems that enforce subsumption between documents (Weaver et al, 1989), making them unsuitable for the complex interrelationships extant in litigation support domains.

5 Querying and Retrieval for Browsing

Generally, it is not sufficient to retrieve a single document that best satisfies some query or is most likely to be relevant to a particular issue in a particular manner. It is more useful to construct a set of documents that are more or less likely to serve the purpose of the user, and then allow the user to browse through this set in a structured manner that has semantic significance. As such, a user query is treated as the means by which the user specifies the *context* in which the relevance of (and similarity between) the documents to be retrieved for browsing is judged.

The user may specify a complex query consisting of a Boolean combination of query elements. Each query element must have the same general structure as relevance functions: ie., consist of one issue and one or more primitive domain concepts. For a document to be retrieved as part of the browsing set, each query element must be matched (or a match excluded where Boolean NOT qualifies the query element) with at least one RF in the document. We will now describe what we mean by a *match* between a query element (referred to below as 'query', for short) and a RF. For a trivial match, i.e. one not relying on explanatory links, the following conditions must be true:

(a) Either the issue specified in the query must be identical to the issue contained in the RF (i.e., both must be instances of the same object in the issues taxonomy); or the two issues must be instances of class objects that share a common parent or grandparent in the issues taxonomy.

(b) Each of the primitive concepts in the query must either be identical to a primitive concept in the RF, or share a parent with that primitive concept.

A non-trivial match can be established by looking at the explanatory links attached to the document to which the RF belongs. For example, consider the situation partly illustrated in Figure 9. There, the two issues to be matched are neither identical nor share a grandparent in the issues hierarchy (Figure 2) because they are in no globally applicable relationship according to the conceptual analysis of the domain. However, a match may be found if the document contains an explanatory link stating that 'loss by extraneous factors' is an *alternative interpretation of* 'advised by third party'. It is necessary to match 'document with disputable provenance' with 'phone log', and 'third party adviser' with 'accountant'. These trivially match, see the taxonomies in Figures 4(a) and (b). However, if a common parent or grandparent could not be found for two primitive domain concepts, but if there was an appropriate explanatory link, say, *implied by*, between the two primitive domain concepts or between one of them and a parent of the other, a match might still have been found.

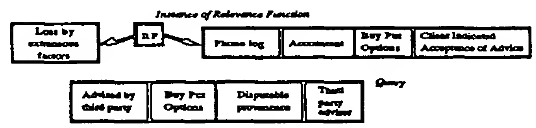

Figure 9 : Matching Query Element with RF

A RF may be said to partially match a query if their issues are matched but one or more primitive domain concepts in the query do not find a match in the RF. Once a set of documents is retrieved, the user can browse through the documents in the order of the degree of matching. Essentially, the system is aiming to provide dynamic clustering of document in accordance with the similarity between them, similarity being judged dynamically in the context of the query, rather than being a fixed parameter. The user can also traverse along the reference links specified in the retrieved documents in order to find other documents which are explicitly referenced or incorporated in, or rebutted by, the retrieved documents.

6 Discussion

6.1 The Relevance Function as a Relatively General Index

Using a function consisting of relationships between certain salient features to index documents is not new. However, most current research is reported to be based on matching only one relation per frame, and "*there is potential ... for considerably improving these methods by allowing matches on more than one relation at a time*" (Lewis et al, 1989). We believe that our work goes some way in this direction by allowing a number of functions to be specified within a document object and then retrieving on the basis of the combined effect of the functions.

We also recognise that it is necessary to ensure that any function is designed so as not to be overly sensitive to minor inconsistencies or variations between the ways in which different encoders view or represent the same concepts. There are several factors in the architecture which contribute to the achieving of this aim.

Firstly, the function is not purely artificial or mathematical; it carries a semantic significance in the domain. The encoders are asked to do little more, at the conceptual level, than exercise their existing faculty of making assessment of the relevance of documents to issues in contention. This they are quite able to do. It is explaining the assessment "through a logical chain of inference" (Ashley, 1989), or decomposing concepts into subconcepts and specifying their relationships and dependencies (as would be required if, say, conceptual dependencies were used to represent the text content of documents), which is difficult. Using existing skills also means that there will be fewer problems with the consistency or integrity of encoding.

Secondly, an exact match between the primitive domain concepts associated in an RF is not insisted upon and the matching can be partial. Moreover, primitive domain concepts that are closely related in the taxonomy are said to match, allowing the encoder some leeway. Where there is doubt about the interpretation, explanatory links can be employed to reflect, to a certain extent, the nature and scope of the doubt.

6.2 Relationship to Functional Similarity in Case-Based Design

It is not necessary here to go into the details of CBD (or case-based planning, which is equally relevant for present purposes); they are elaborated elsewhere (Hammond, 1988; Mital 1990). Essentially, for every problem-solving step, a CBD system searches for a case that deals with a situation that is 'similar' to the problem situation at hand. Determining that two situations are similar is a crucial step in the drawing of an analogy. The process of making analogies between two states of affairs allows us to infer from the fact that there are some similarities between the states that there must be other similarities (Leishman, 1990) - i.e., that the step employed in the retrieved case is applicable to the current problem situation. In this sense, a similarity between two situations is a commonality at some level of abstraction. Of course, establishing (or even defining) similarity may be very complex in the case of "without-domain" or inter-domain analogy. For instance, a student learning about heat transfer can map the knowledge that water falls from a high elevation to a lower one into the heat transfer domain, and from that derive an understanding as to the direction in which heat flows between two bodies at different temperature levels. There, complex issues such as systematicity are involved (Gentner & Toupin, 1986). Fortunately, we are dealing strictly with 'within-domain' analogy, where all concepts to be considered belong to the same domain and it can be taken that identical predicate structures have the same sematic significance throughout. In such a situation:

> "*Object similarity can potentially be reduced to predicate similarity: two objects are similar to the extent they serve as arguments of similar predicates.*" (Holyoak & Thagard, 1989).

However, searching for objects which are similar to the problem situation can still be computationally expensive. It is necessary to ensure that the indices bear a close relationship to the particular notion of similarity employed in a system (Mital, 1990). Also, that the features indexed upon are not such as to exist in the domain in very large variety. In CBD, it has been pointed out that while a huge variety of actual 'output behaviours" of design cases exist, the '*desired* output behaviours' are limited in number (Goel, 1989). In choosing as indices features indicating the purpose (seen from the user's point of view) of documents, rather than the actual combination of concepts occurring in the text, we are acting accordingly.

6.3 Part of a wider effort

The system which has been discussed above is part of a concerted, broader approach to information management for practice support being taken at Brunel University. Additional areas of research include using hypertext for legal document assembly (Southam et al, 1991) and neural networks for automatic text analysis and information retrieval (Gedeon & Mital, 1991).

7 Conclusions

One of the central strands of current artificial intelligence research involves adapting, refining and augmenting existing techniques to suit particular, well-defined domains and applications. This is as a consequence of the recognition that the search for general purpose representation schemata and inferencing mechanisms has left behind significant gaps that need to be filled *de novo* every time a practical development is carried out. We have shown that the litigation support application - one of enormous commercial importance - has peculiar characteristics that necessitate the use of special techniques. These characteristics become apparent only when the specific nature of the application is thoroughly investigated, rather than through an analysis of the nature of legal concepts in general.

References

Ashley, K.D. (1989) "Indexing and analytic models." *Proceedings, DARPA Case-Based Reasoning Workshop*, 197-202. Bareiss, R. (1989) *Exemplar-Based Knowledge Acquisition*, San Diego, Academic Press.

Belew, R.K. (1987) "A connectionist approach to conceptual information retrieval." *Proceedings of First International Conference on Artificial Intelligence and Law*, 116-125.

Bertaina, P.; Borello, E.; Di Leva, A. & Giolito, P. (1982) "The architecture of a legal information system." in C. Ciampi (ed.) *Artificial Intelligence and Legal Information Systems*. Amsterdam, North-Holland.

Berul, L.H.; Stern, M.P. & Keane, J. (1980) "The need for computer-managed litigation files." in B.K. Eres (ed.) *Legal and Legislative Information Processing*, Westport, Greenwood Press.

Bing, J. (1989) "The law of the books and the files: possibilities and problems of legal information retrieval." in G.P.V. Vandenberghe (ed.) *Advanced Topics of Law and Information Technology*, Hillsdale, NJ, Lawrence Erlbaum, 151-182.

Blair, D.C. (1990) *Language And Representation In Information Retrieval*. Amsterdam, Elsevier.

Blair, D.C. & Maron, M.E. (1985) "An evaluation of retrieval effectiveness for a full-text document retrieval system." *Communications of the ACM*, **28**, 3, 289-299.

Christian, C. (1990) "Litigation support systems - one small step for credibility." *New Law Journal*, November 9, 1990, 1567-1568.

Cornwell, D. (1990) "Innovative applications of full text retrieval". *Computers and Law*, October, 7-8.

Cross, G.R. & deBessonet, C.G. (1985) "Representation of legal knowledge for conceptual retrieval." *Information Processing & Management*, **21**, 1, 35-44.

Dick, J.P. (1987) "Conceptual retrieval and case law." *Proceedings of First International Conference on Artificial Intelligence and Law*, 106-114.

Galloway, T.Y. (1987) "TAXI: a taxonomic assistant." *Proceedings of AAAI-87*, 416-420.

Gedeon, T.D. & Mital, V. (1991) "An adaptive learning network for information retrieval in a litigation support application." *Proceedings of AMSE International Conference on Neural Networks*, San Diego.

Gelbart, D. & Smith, J.C. (1990) "Knowledge-based information retrieval systems." *Computers & Law*, October, 1990, 23-26.

Gentner, D. & Toupin, C. (1986) "Systematicity and surface similarity in the development of analogy." *Cognitive Science*, **10**, 277-300.

Goel, A. (1989) "Integration of model-based reasoning and case-based reasoning for design problem solving." *Ph.D. Thesis*, Ohio State University.

Hafner, C.D. (1981) *An Information Retrieval System*. Ann Arbor, Research Press.

Halverson, J.T. (1979) "Use of computer for manipulating information: a trial lawyer's guide to document storage and retrieval." in J.H. Young, M.E. Kris and H.C. Trainor (eds.) *The Use of Computers in Litigation*, American Bar Association.

Hammond, K. (1988) *Case-Based Planning*, San Mateo, CA, Morgan Kaufman.

Holyoak, K.J. & Thagard, P. (1989) "Analogical mapping by constraint satisfaction." *Cognitive Science*, **13**, 295-355. Keane, J.I. (1989) "The automated litigator." *Trial*, January, 1989, 30-38.

Kim, W. (1990) "Object-oriented databases: definition and research directions." *IEEE Transactions on Knowledge and Data engineering*, **2**, 3, 327-341. Leishman, D. (1990) "An annotated bibliography of works on analogy." *Int. Journal of Intelligent Systems*, **5**, 43-81.

Lewis, D.D., Croft, W.B. & Bhandaru, N. (1989) "Language-oriented information retrieval." *Int. Journal of Intelligent Systems*, **4**, 285-318.

Martin, C. (1989) "Indexing using complex features." *Proceedings of DARPA Case-Based Reasoning Workshop*, 26-30.

Mital, V. (1990) "Case-based design and conceptual information retrieval in domains with a weak theory: financial/legal planning and litigation support applications." *Brunel University Technical Report* no. CSTR-90-7, available from author.

Mital, V. & Johnson, L. (1991a) *Information and Knowledge Systems in Law and Finance*, London, Chapman & Hall.

Mital, V. & Johnson, L. (1991b) "Professional negligence and financial-legal expert systems: architectures to enable the reasonableness defence." *International Journal of Law and Artificial Intelligence*, **1**, 1.

Mital, V.; Stylianou, A.K. & Johnson, L. (1991) "Litigation support systems: an object-oriented approach to conceptual information retrieval." *International Journal of Systems Research and Information Science*, to appear.

Patel-Schneider, P. (1990) "Practical, object-based knowledge representation for knowledge-based systems." *Information Systems*, **15**, 1, 9-19.

Rau, L.F. (1987) "Knowledge organization and access in a conceptual information system." *Info.Processing & Management*, 23, 4, 269-283.

Rissland, E.L. & Ashley, K.D. (1989) "HYPO: A precedent-based legal reasoner." in G.P.V. Vandenberghe (ed.) *Advanced Topics of Law and*

Information Technology, Hillsdale, NJ, Lawrence Erlbaum, 213-234.

Schank, R.C., Ed. (1975), *Conceptual Information Processing*, Amsterdam, North Holland. Southam, M.; Mital, V. & Thomas, P. (1991) "HyperNotary: Legal document assembly using hypertext." *Proc. 13th BCS Information Retrieval Colloquium*, Lancaster.

Stylianou, A.K. (1990) "An Object-like Knowledge Representation Language (OKRL) for knowledge and information representation." *Brunel University Tech. Report* no. CSTR-90-9.

Tong, R.M., Appelbaum, L.A. & Askman, V.N. (1989) "A knowledge representation for conceptual information retrieval." *Int. Journal of Intelligent Systems*, 4, 259-283.

Wallwork, N.J. (1989) "Managing documents in complex litigation." *Legal Economics*, November/December, 1989, 52-57.

Weaver, M.T., France, R.K., Chen, Q-F & Fox, E.A. (1989) "Using a frame-based language for information retrieval." *Int. Journal of Intelligent Systems*, 4, 223-257.

Wilkins, R.P. (1989) "Computerizing your litigation practice." *Trial*, April, 1989, 82-83.

1.Another line of research is predicated on the twenty five year old assumption that the formal aspects of text, such as the relative frequencies with which a particular word occurs in a particular document and in the document database in general, can predict the meaning or subject content of the text (Blair, 1990). This line of research is of more relevance to the connectionist approach to artificial intelligence (Belew, 1987; Gedeon & Mital, 1991) than to the symbolic processing stance implicitly taken in this paper.

2.The litigation team may have first tried to deal with the information by manual cataloguing and indexing, until the problems became overwhelming (Berul et al, 1980). The possibility of an early settlement may have been in the air (though, some enlightened litigators use LSS as an aid to settlement itself (Keane, 1989)). Discovery from multiple parties may have taken place asynchronously or at a late stage.

3.More correctly, an instance of the explanatory link class object relates instances of objects in the issue and primitive domain concepts class lattices. A similar comment will apply to the description of reference links.

4.There is no global theory of relevance of concepts to issues in the domains likely to be litigated about and universally applicable relations are difficult to state(cf. Ashley (1989)).relevance of concepts to issues in the domains likely to be litigated about and universally applicable relations are difficult to state (cf. Ashley (1989)). Yet, given the facts of a situation, it is possible for a lawyer or a paralegal thoroughly familiar with the case, to usually say that a document is likely to be relevant to a particular issue, and that it is so relevant because of the presence in the document of references to certain concepts.

5.Actually, rather than linking and relinking RF's (which are complex objects, much memory/storage management would be needed for reorganisation), the discrimination is done by means of special objects called RF-skeletons. Every time an RF is specified a corresponding RF-skeleton object containing *attributes* that are equivalent to the *values of the attributes* of the RF is created: the rectangles in Figure 8 represent RF-skeletons which have been so created. The discrimination algorithm involves checking whether there exists in the hierarchy a RF-skeleton class object which has attributes such as to subsume the new RF-skeleton. If it does, the new RF-skeleton is made an instance of the existing class object: in Figure 8, the class object DIS subsumes two new RF-skeletons. If no such class exists, a new class is created (the user being consulted as to the location of the class where more than one location is possible). Further details are given elsewhere (Mital et al, 1991). RF-skeleton class objects accumulate a list - not shown in Figure 8 - of unique identifiers of all those documents which contain RF's with identical attribute values. By extension, by means of siblings in the hierarchy, we can also have direct access to those documents which contain RF's with only one attribute differing in value, or two, and so on.

Search Results

Search Results for: **[image<AND>(((diferent <near/2> (type or format)) and (different <near/2> (files or types))))]**
Found **2** of **127,944 searched.**

Search within Results

> Advanced Search

> Search Help/Tips

Sort by: Title Publication Publication Date Score Binder

Results 1 - 2 of 2 short listing

1 Geologic hypermaps are more than clickable maps! 87%
Agnés Voisard
Proceedings of the sixth ACM international symposium on Advances in geographic information systems November 1998

2 A general method for spatial reasoning in spatial databases 77%
Alia I. Abdelmoty , Baher A. El-Geresy
Proceedings of the fourth international conference on Information and knowledge management December 1995

Results 1 - 2 of 2 short listing

Geologic Hypermaps are more than Clickable Maps!

Agnès Voisard*
Computer Science Institute
Freie Universität Berlin
D-14195 Berlin, Germany
voisard@inf.fu-berlin.de

Abstract

Geologic maps are interpretations of 3-D phenomena. The major aspects to be modeled in such maps are fuzziness (both on assumptions and on geometric components of the geospatial objects they contain) and complex relationships among the underlying data. In addition, map making is an incremental process which asks for multidimensional versioning on spatial components, time and assumptions. Hypermaps are composed of maps, multimedia objects and links among these objects. In this paper we first identify the database features needed to model geologic hypermaps. We then propose a query classification based on data types as well as a model to describe the structure of such maps.

1 Introduction

A geologic map provides information regarding the underlying structures of a given area. It contains many different types of information, which makes it complex. It is based on various types of information such as field observations, drill holes and fossil records. From this information the geologist develops a conceptual model describing the geologic processes that shaped the observed phenomena. Such a model is represented by a geologic map with which is associated explanations, legends, geologic profiles, photos and base data. Spatial parts of such maps are stored as vector objects. Other types of maps, namely images, stored as bitmaps (raster) are also considered. They are often used as backgrounds of maps on the screen.

A geologic map is hence defined according to a certain knowledge of the fields and hypothesis on given areas. Even though the output (the map) is frozen, it is useful to know which hypothesis from which expert lead to a certain ouput, e.g., which assumptions allowed a map maker to say that in this area the soil concentration in iron is higher than the one of cobalt. This may be based on geologic models whose description must be accessible to endusers. So far, this knowledge, associated with a given map, was given by textbook. With the advent of GIS, it should be accessed on the screen via a rich map able to convey a large amount of heterogeneous information. In addition, the same geographic space could be associated with other assumptions that could be accessed as well, either from the geographic data themselves on the screen or even from related assumptions. This is very similar to hypertexts and hyperdocuments. Hypermedia techniques [1] are obviously well-suited to handle such maps graphically and interactively. Assuming such maps are made of geologic objects, from a map displayed on the screen, endusers can click on a point of a map and get (i) information regarding assumptions that led to the definition of a certain object and (ii) characteristics of this object.

Although the area of hypertext has received lots of attention in the past 20 years the concept of hypermaps is, to the best of our knowledge, rather new and not well-understood. Part of the challenge is that a data access based on coordinates (geographic aspects) has to be provided. A description of some of the problems encountered when dealing with hypermaps can be found in [12]. The idea is to extend the hyperdocument concepts by integrating geographic referencing. The hypermap is defined in terms of semantic units, and links allow one to access information within a semantic unit or in a related semantic unit (hence information is organized in co-webs with nodes). In traditional hypertexts, links are defined from documents to documents ("text-to-text"). The novelty of the hypermap concept involves the consideration of new types of objects as targets. Hence links such as map-to-map, map-to-image, image-to-map, map-to text, image-to-text ("hyperbitmaps",i.e., raster with sensitive zones) and all possible combinations are also part of the system. Moreover, the nature of links towards elements of geologic maps go beyond those for traditional hypermedia applications. For instance, special links regarding assumptions have also to be part of such systems.

Geologic map making is an incremental process. A geologist usually starts with information regarding a given region in the form of digitized maps. Such a map is refined with a better knowledge of the field (using drill holes, for instance), images as well as new understanding (interpretations, theories) of geophysic phenomena. Sometimes assumptions are not valid anymore. In some circumstances objects have to be deleted or new objects have to be added. Explanations regarding these updates are also part of a geologic map. This shows that even though a geologic map is not a pure dynamic object, it is likely to evolve with time and it con-

*Part of this work was financed by the German Research Foundation (DFG), within the IOGIS framework on interoperable GIS.

ACM GIS '98 11/98 Washington, D.C., USA
© 1998 ACM 1-58113-115-1/98/0011...$5.00

tains an increasing amount of information.
As far as time itself is concerned, it does not play a crucial role such as in some other geospatial applications (e.g., transportation domain). In the sequel it is handled as a regular attribute. However, time as metadata (for map versioning) has to be encompassed in a geologic hypermap model.

Users of these systems are both map makers (database designers) and endusers. While a map maker is interested in operations such as the evolution of map (interpretation of repeated phenomena, for instance) and the production of a new version of the map as seen above, the naive enduser, for instance, an engineer who only wants to install a pipe in a soil, will be concernned with a frozen representation of a given map. A more sophisticated enduser such as a geology student would like to find out which assumption(s) lead to which definition of geologic objects. Our goal is to study a model and query language for both the designer and the enduser (note that the designer, because of the incremental map making process, is also a sophisticated enduser). A desired query language must allow access by both contents and structure.

It is our belief that geologic hypermaps will greatly facilitate the life of geologic map makers and will change many aspects of map making, not only by clicking on maps and documents. In fact, because map making is a step-by-step process, many versions of a map designed under different assumptions can be stored and gathered in appropriate structures. Retrieving information related to different versions of a map is part of map understanding. In other words, keeping map evolution, which was extremely tedious before such maps where computerized, is an important component of the geologic map making process. To our knowledge, even though there are some attempts to model a somewhat frozen geologic map [13], nothing has been done yet in that direction.

This work was carried out within a joint project with geologists. The first step was to understand their needs and to realize a prototype. The prototype was based on a simple map/assumption model using ArcView [11] and its programming language Avenue [7]. The underlying model was unfortunately not powerful enough to handle the complex requirements of geologic hyperapplications. The second phase of the project is geared towards the definition of a powerful data model and query language to handle many situations.

This paper is organized as follow. Section 2 presents the major requirements of such applications. We describe the objects of interest as well as representative queries. Section 3 presents a geologic map model based on hypergraphs.

2 Understanding Geologic Maps

For a neophyte like a computer scientist in our case, trying to understand all concepts of a foreign family of applications is a challenging task that relates to cognitive aspects. During the first phase of our project we got a good insight into the main features and problems of geologic applications. This section first describes data, metadata and relationships among them in such applications. We then elaborate on querying the underlying databases. Different aspects of fuzziness arising when designing geologic hypermaps are then exposed. A list of requirements for such applications concludes this section.

2.1 Data and relationship among them

Base data

- **Maps and geo-bjects.**
A conventional geologic map is an interpretation of observed phenomena. It is built from (i) individual maps such as geologic layers, hydrology, faults, soils, topography, (ii) explanations (textbooks) as well as (iii) additional material: Data from drill holes, aerial photographs, drawings, notes from the fields, geochemic data, etc. The idea is to combine these different types of information within a coherent user-friendly framework that can be manipulated interactively, namely a geologic hypermap.

  A complete geologic hypermap is composed of thematic layers, or thematic maps[1], where each layer is a collection of geographic or geologic objets. In the sequel, for the sake of simplicity, we refer to all these objects as geo-objects unless we have to address geologic objects explicitly. A geo-object is defined as a tuple: [alphanumeric description, spatial component].

  The spatial part is concerned with both geometric and topologic aspects. It is usually modeled as an abstract data type. Point features include sample location symbols (drilling points). Linear features, or 1-dimensional objects, are for instance faults, folds or dikes. Polygonal areas, or 2-dimensional objects (e.g., lithologic units), are more complex than in traditional GIS applications, as the lines separating polygons has a special significance in geology (type of contact between two geologic features). In this paper we do not focus on the sophisticated modeling of the geometry of geologic objects. Information on this topic can be found in [3].

  In addition, the structure of geo-objects is complex as an object can be composed of other objects (e.g., a river composed of branches). Often, the underlying geometry form a partition.

  As far as spatial components are concerned, the imprecision of the terrain induces a fuzziness in the geometry of the geo-objects, as we will see later. Since many objects are defined as being adjacent within a layer (e.g., a fault between two fields), topological models (see [6]) are more suitable for representing these objects than a pure geometric model such as a spaghetti model. Users interact with maps displayed in windows. In the current interface model we use the notion of Mapget [14], which is a window devoted to querying and editing. Its main characteristics is that it contains layers of information, which are organized in stacks with an active layer on top of the stack. This layer can be modified and queryied while the other ones are only visualized. Other operations on sets of layers are also defined.

- **Multimedia objects and documents.**
Images (bitmaps) can be displayed either separately or as backgrounds. Photos, videos are attached to certain objects. Textual objects are typically parts of textbook and they represent descriptions, explanations as

[1] The term "map" may not seem appropriate to the GIS community where map usually denotes a fixed representation (on the screen for example) of what is denoted map here.

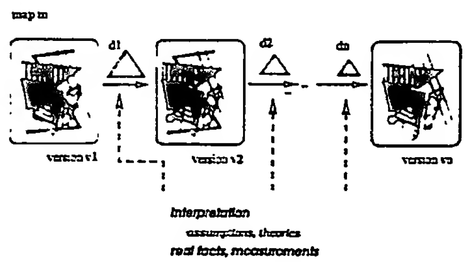

Figure 1: Illustration of the map making process

well as assumptions. As explained further they are linked to geo-objects and to other parts of textbooks.

- **Cartographic objects.**
  The legend plays a crucial role in such applications, not only to convey rapidly information to the enduser, but especially because of the lack of standards regarding symbolization in that disciplin. A particular designer will use common representation but also his/her own symbols. Associated with each geo-object on the final map is a legend (the representation of a geo-object can be seen as a join between a geo-object and a cartographic chart of a given interpretation). In the sequel we do not focus on these cartographic aspects. For more information regarding this topic, such as the distinction between many types of legend, see [14].

Relationships among objects

The main relationships among objects are historic, semantic associations and hypothetic associations as described thereafter.

- **Keeping track of histories.**
  Map making is a step-by-step process. The geologist starts with a given version and then enriches it thanks to a new understanding of phenomena (based for instance on field measurements or even on recent more abstract theories). Figure 2.1 illustrates the process evolution. Note that the difference Δ between two maps tends to be smaller and smaller. Understanding geologic maps relies partly on the *understanding of the evolution* of a map of a given area. This is also a reason why a computerized-version of geologic maps (instead of archiving all paper versions) will probably influence greatly geologists' methods of work in the near future.

- **Associations among objects.**
  It is useful to associate some objects with each other, for instance when they have geologic explanations in common. These objects will be part of semantic units.

- **Assumptions on objects and existence dependencies.**
  So far, the points above are not far from the characteristics of geospatial data in general. However, considering geologic maps brings more complexity as some geologic objects exist only because

  - either assumptions were made on them.
  - or other objects exists. For instance, the presence of a fault introduces a partitioning of polygons. A polygon P_1 may be split into the following collection of objects: (f, P_2, P_3) where f represents the geometry of the fault.

Regarding the later case, if an assumption on these objects is not valid anymore, or if the objects associated with them do not exist any longer, then their existence is not justified anymore. Objects whose existence depends on assumptions are similar to geo-objects in general, but a link towards an assumption (text) or a set of assumptions is attached to them. Objects whose existence depends on the existence of other objects have to be defined differently. In a data model they could rely on a special set-similar constructor.

Metadata

In geospatial applications metadata refer to data describing the base data, such as all global data about the geospatial objects that can be factorized (origin of data, date, classifications, etc.). Note the difference with the meaning in databases, where data refer to the structure of the data. remark Such applications require an explicit modeling of metadata [9]. In our application, the metalevel includes the following features:

- Legend: Graphical representation attached with all displayable objects. This is the global variable of an application as it ensures the consistency of the display.
- Links that exist in the database.
- Annotations on sets of objects, such as the date and the origin of data (drill holes, surveying) as well as the author.
- Categories of objects for classification (type of soil, etc.).

2.2 Querying

In geologic map systems, naive endusers typically pose queries interactively (using a mouse device) against the database in order: (i) to identify objects or parts of objects or (ii) to select graphical parameters for spatial queries. More sophisticated users need to access more elaborate knowledge on the contents of data. Designers are usually more interested in the structure of data. In the sequel we give example of queries on map contents as well as map structures.

On map contents (base objects)

These are basically operations of the relational algebra.

- on GeoObjects (spatial/alphanumeric parts)
  - Q1. *What are all the alphanumeric properties (description) of this object?* (relational selection in standard databases)
  - Q2. *What are all geo-objects in this area?*
- on textual objects
  Q3. *What are the parts of textbook TB1 where "iron" is mentioned?*

- on assumptions
  Q4. *What are all the assumptions of Ms. XYZ in this area?*

- on semantic units
  Q5. *In a given map version what are the objects of Layer "Hydro"*

- on Maps
  Q6. *What are the maps designed after October 1990?*

- on (geo)meta data such as legend and annotations
  Q7. *What is the way Ms. X represents a portion of soil containing gold?*
  Q8. *What are the existing classifications of soils in the database?*

- on linked GeoObjects
  Q9. *what is all available information on this object?*

On the structure

- on linked GeoObjects
  Q10. *what is the type of available information on this object?*
  Q11. *What objects are related to this one?*

- on links
  Q12. *what assumptions led to the definition of this object (if any)?*

- on sets of GeoObjects
  Q13. *Which objects are not based on assumptions?*
  Q14. *Which objects are defined under this assumption?*
  Q15. *Which objects are defined exclusively under assumptions?*

- on Maps
  Q16. *What maps were defined with geologic map GM as a starting point?*
  Q17. *What are the maps using assumption A?*
  Q18. *How many versions do I have for a geologic map CGM?*

On both contents and structure

This category also contains *what-if* queries which are common tools for simulation in spatial decision-support systems as illustrated Query Q19.

- on assumption link with contents of assumptions known
  Q19. *What if assumptions of Ms. XYZ is not justified anymore?*

- on assumption links with contents of assumptions unknown
  Q20. *Which objects are defined exclusively under (which) assumptions?*

2.3 Different facets of fuzziness

A geologic map being a geographer's interpretation of the real world, together with the fact that many design steps are empirical, a lost of information, as well as uncertainty or fuzziness are obviously introduced during map making process. Below is a brief description of the types of fuzziness to be handle in these systems.

- **On assumptions.**
  The hypothesis on which the assumptions are based constitute an important aspect of fuzziness. Such hypothesis much appear explicitly with a coefficient of uncertainty.

- **Missing information: Need for interpolation.**
  Major features needed for making geologic map are drilling points in the fields. From these points geologists get information at precise coordinates. Interpolation is needed to infer values between points. Interpolated values are intrinsically uncertain. They lead to the definition of geo-objects, and they are written in the database in a text form. In addition, the accuracy of the values measured at the drilling points cannot be taken as granted.

- **On the geometry.**
  Between areas are fuzzy borders. This induces a fuzziness in the geometry of geo-objects. This topic is out of the scope of this paper. Approaches such as the vague regions described in [8] are good candidates to handle this problem.

- **On geo-objects in general.**
  In addition, a map covering a large area is composed of many submaps. This asks for seamless integration of data sheets. This is a complex requirements as map providers do not necessary agree on coordinates of geospatial objects. It introduces fuzziness between different data sheets.

2.4 Summary: Requirements for geologic hypermaps

To sum up this introduction to geologic maps, below are the features that need to exist in a generic hyper geologic map model:

- Basic extensible geologic map model with geologic and multimedia objects, which encompasses the notion of complex object and of "if-objects".
- The possibility of grouping objects together.
- The explicit representation of assumptions with explanations.
- Different types of links among objects.
- A possibility of versioning (indexed on time and assumptions) as map making is an incremental process.
- The expression of uncertainty at many levels of representation.
- A query language that (i) accesses both the structure and the contents of maps, (ii) allows recursion and (iii) allows negation.
- A way to customize the underlying structure and to tailor it to one's specific needs.

Taken separately, many tools are of interest to model the complex requirements of such applications. For instance, in the logic domain, intepretations, domains and models are obviously well-suited to handle the different interpretations of a given area. Deductive databases are of great interest to infer new facts. Versioning plays an important role although it should not be based on time only such as in

standard database applications. Petri nets are also candidates for handling assumptions. Truth maintenance systems (e.g, ATMS) are well adapted for handling hypotheses that might be falsified. They encompass a simultaneous consideration of many different interpretations ("possible worlds"). Each world or context is consistent, but they are inconsistent among each other. Such systems focus on inference and help to find out the valid conclusions that can be drawn if a hypothesis is verified or falsified.

As far as the underlying structure is concerned, graphs are are obviously the right tools to model the web among related objects However, in geologic applications, a few extensions should be introduced to consider richer relations than pure hyperlinks. Moreover, the choice of associating a given feature or piece of information either with objects or with links results in the following trade-off: If nodes (objects) carry lots of information, this information is easy to access but the web ends up being very large. On the other hand, if nodes are kept small, the information will be concentrated on edges which makes sophisticated navigation a primordial goal. Next section presents our compromise on this aspect.

3 A Geologic Hypermap Model and Query Language

This section first describes our basic graph model. This model was translated into an O_2 schema in a long version of this paper [15], and the queries of Section 2 were expressed in the O_2Query language. We conclude this section with remarks on the navigation and querying in such structures.

3.1 Basic Model

Definition 3.1 A complete geologic hypermap map is a tuple (i, c, VG), where (assuming S an infinite set of strings)

1. i is an interpretation defined as a pair (a, t) where a is a set of strings (an author or a group of authors) and t a theory (e.g., a reference to a geologic model). $dom(i) = \{\{S\} \times S\})$
2. c the coordinates of the area ($dom(c) = (\Re \times \Re)$)
3. VG (version graph) is a tree (V_{GHM}, E_{GMH}, H). Vertices V_{GHM} are geologic hypermaps defined below. E ($E \subseteq (V_{GHM} \times V_{GHM})$) is the set of edges that link geologic hypermaps. H ($H : E \to Text$) is a set of parametrized (historic) link among versions, with parameter of domain Text (explanation on the transition from map v_1 to map v_2). Type Text is a complex type made of components of type string (to represent structured documents).

Definition 3.2 V_{GHM} is a directed weakly connected graph (V_{HO}, E_{HO}, F), where

1. V_{HO} is a set of hyperobjects HO.
2. E_{HO} is a set of edges $E_{HO} = (E_{HO_A} \cup E_{HO_S})$, where $E_{HO_A} \subseteq HO \times HO$ and $E_{HO_B} \subseteq HO \times HO$.
3. $F : E_A \cup E_S \to \mathcal{P}(V_{HO})$ is a parametrized incidence function.

Three remarks are noteworthy:
1. The situation with two same edges going from node HO_1 to node HO_2 cannot be considered by the graph above. However, such situations arise only with graph customization (many customizations leading to the same graph structure). It is easily handled with subgraphs.
2. Nothing prevents an hyperobject to have assumption link to itself in the structure above, i.e. the situation with edge $(ho, ho) \notin E_{HO}, ho \in HO$.
3. An assumption subgraph G_A (resp. semantic subgraph G_S) will be defined as $(V, E_A, F : E_A \to \mathcal{P}(V_{HO}))$. Customized environments will also be created by extracting relevant subgraphs.

3.2 Wishlist for geologic hypermap query languages

One of the main criticisms regarding the use of basic hypermedia techniques in the GIS context [4] is that queries are already defined in the sense that (hyper)links have been established between entities beforehand. This means that paths are pre-compiled and that users cannot ask sophisticated queries. Another criticism is that underlying data models are usually extremely poor, with the sole concept of (hyper)links among entities.

The queries above show that navigating in such structures is straightforward when there is no recursion. In such environments recursion is handled by embedding an SQL-like language in a multi-purpose programming language which allows loops. In addition, queries accessing objects types (the schema) cannot be handled in one step as illustrated by Query 12b. Query Q19 of Section 2 (*What if assumptions of Ms. XYZ is not justified anymore?*) has to do with the creation of a new map based on the "undo" operation. The idea is to extract all objects defined under a given set of assumptions (Cf. Query Q14), and to remove them from a map. Note that this has to be done in many steps (recursion). As far as Query Q20 is concerned, it could not be expressed either as, similarly to Query Q12b, it needs access to both the schema and the instances simultaneously.

The problem is to find a query language that embeds both aspects in a consistent framework. Such a query language must support negation and recursion on the structure as illustrated above. In addition, because of the lack of experience in querying of some naive users, expressing queries should be simple. Languages such as GraphLog [5] and all its derived languages seem well-adapted to the user-friendly graph manipulation. GraphLog is a visual query language in which queries are formulated by drawing graph patterns. It allows the specification and manipulation of arbitrary subsets of the network (also useful for customization) and supports the computation of aggregate functions on subgraphs of an hyperdocument. However, it does not support the 2-level querying needed in our case.

The peculiarity of our case is that we know in advance the types of objects and nedges that we manipulate, even though they can all be specialized for customization. In that sense most of the tools like [5, 2, 10], which allow to define any relationship among objects, are extremely rich as far as our graph modeling is concerned.

In addition, we need to consider a special "undo" operation in case an assumption is not valid any longer. This operation creates a new geologic map (remember that we want to keep track of the evolution of a map) in a com-

plete geologic map and introduces an instance of history link. Suppose that map gm is the union of the set SGA of geo-objects relying on assumption A and other objects not relying on A (set SG-notA). The new map is populated with elements from set SG-notA only, which implies the existence of a recursive function able to retrieve all geo-objects defined under hypothesis A. Other assumptions will be made further to create new maps.

4 Conclusion

New geospatial applications need the consideration of complex interactive objects, namely hypermaps. The idea is to move away from the classical paradigm where maps (sets of geospatial objects) are just displayed on the screen in the form of cartographic objects, and geospatial objects' description accessed through a mouse click. Hence a map tends to become more than a somewhat static object on the screen: It is rather a dynamic and interactive object that allows one to acces related information. Indeed, in many geographic applications, a knowledge other than pure cartographic/geo-based output has to be accessed by endusers who may want to access documents from geographically-referenced data.

Geologic hypermaps are a new generation of maps based on such mechanisms. They are meant to be used by both endusers and map makers. With a rich querying mechanism on different types of documents they obviously make endusers' life more comfortable. The major aspects of such maps is the new potential offered to designers (map makers). We insisted on the fact that understanding a geologic map evolution is part of the understanding of geologic maps and phenomena. On this aspect, geologic applications differ very much from other geospatial applications. We believe that techniques such as the ones presented in this paper are likely to change radically the geologic map making process.

This work was based on a first study that led to a prototype implementation on top of the GIS-interface generator ArcView [11, 7] from ESRI, Inc. Our complete geologic map covered an area in Sauerland, Germany (scale 1:100 000). Even though the user interfaces aspects are extremely sophisticated in this software (complex graphic object types, charts, etc.), this experiment showed us limits of such tools as far as database features are concerned. For instance, modeling typed links was done through hacks in scripts. Due to data modeling limits, our prototype was also base on a simple model for handling maps and assumptions. Now that we got insight into geologic problems, the second phase of the project is to defined a generic map model, on which we briefly reported in this paper.

In this paper, we first described the entities and relationships to consider in such applications. We also gave a short description of the types of fuzziness to be taken into account and we gave a taxonomy of queries on such systems. Understanding the requirements of such complex applications, which is time-consuming for a computer scientist, is a challenging task. We then presented a model based on graphs, as it naturally comes to mind. The model defined here is based on a simple 2-level graph model that allows versioning. In this paper, we restricted our attention to standard versioning, but it became clear that multidimensional versioning on time, spatial components and assumptions should be part of such systems.

References

[1] ACM and Association for Computing Machinery, editors. *Communications of the ACM, Special issue on Designing Hypermedia Applications*, New York, August 1995.

[2] B. Amann and M. Scholl. Gram: A graph data model and query language. In *Proceedings of the Fourth ACM Conference on Hypertext*, Theoretical Foundations, pages 201–211, 1992.

[3] O. Balovnev, M. Breunig, and A. B. Cremers. From GeoStore to GeoToolKit: The second step. In *Advances in Spatial Databases, Lecture Notes in Computer Science*, volume 1262, pages 223–238, Heidelberg/Berlin/New York, 1997. Springer Verlag.

[4] P. Boursier and M. Mainguenaud. Spatial query languages: Extended SQL vs. visual languages vs. hypermaps. In *Proceedings of the 5th International Symposium on Spatial Data Handling*, volume 1, pages 249–259, 1992.

[5] M. P. Consens and A. O. Mendelzon. Expressing structural hypertext queries in graphlog. In *ACM Hypertext'89 Proc.*, Inf. Retrieval II, page 269. 1989.

[6] M. Egenhofer. A Topological Data Model for Spatial Databases. In *Design and Implementation of Large Spatial Databases*, page 271, Berlin/Heidelberg/New York, 1989. Lecture Notes in Computer Science No. 409, Springer-Verlag.

[7] Inc. Environmental Systems Research Institute. Avenue: Customization and Application Development for Arcview, 1996.

[8] M. Erwig and M. Schneider. Vague regions. In *Advances in Spatial Databases, Lecture Notes in Computer Science*, volume 1262, pages 223–238, Heidelberg/Berlin/New York, 1997. Springer Verlag.

[9] O. Günther and A. Voisard. Metadata in Geographic and Environmental Data Management. In W. Klas and A. Sheth, editors, *Managing Multimedia Data: Using Metadata to Integrate and Apply Digital Data*. Morgan Kaufmann, 1998.

[10] T. Isakowitz, E. A. Stohr, and P. Balasubramanian. RMM: A methodology for structured hypermedia design. *Communications of the ACM*, 38(8):34–44, August 1995.

[11] J. Dangermond. ArcView Status and Direction. *ESRI Arc News*, 16(2):1–6, 1994.

[12] R. Laurini and T. Thompson. *Fundamentals of Spatial Information Systems*. The A.P.I.C. Series, Number 37. Academic Press, 1992.

[13] US Geological Survey. Digital Geologic Maps Data Model, 1997. URL = http://www.usgs.gov/.

[14] A. Voisard. Mapgets: A Tool for Visualizing and Querying Geographic Information. *Journal of Visual Languages and Computing*, 6, 1995.

[15] A. Voisard. Geologic Hypermaps are more than Clickable Maps!, 1998. To appear as a FU Berlin technical report, fall 1998.

Search Results

Search Results for: **[(diferent <near/2> (type or format)) and (different <near/2> (files or types))]**
Found **5** of **127,944 searched.**

Search within Results

> Advanced Search

> Search Help/Tips

Sort by: Title Publication Publication Date Score Binder

Results 1 - 5 of 5 short listing

1 Geologic hypermaps are more than clickable maps! 100%
Agnés Voisard
Proceedings of the sixth ACM international symposium on Advances in geographic information systems November 1998

2 A general method for spatial reasoning in spatial databases 100%
Alia I. Abdelmoty , Baher A. El-Geresy
Proceedings of the fourth international conference on Information and knowledge management December 1995

3 A semantic meta-modelling approach to schema transformation 100%
Mike P. Papazoglou , Nick Russell
Proceedings of the fourth international conference on Information and knowledge management December 1995

4 Experiments in designing computational economies for mobile users 99%
Tracy Mullen , Jack Breese
Proceedings of the first international conference on Information and computation economies October 1998

5 Type specialisation for imperative languages 99%
Dirk Dussart , John Hughes , Peter Thiemann
ACM SIGPLAN Notices , Proceedings of the second ACM SIGPLAN international conference on Functional programming August 1997
Volume 32 Issue 8

We extend type specialisation to a computational lambda calculus with first-class references. The resulting specialiser has been used to specialise a self-interpreter for this typed computational lambda calculus optimally. Furthermore, this specialiser can

perform operations on references at specialisation time, when possible.

Results 1 - 5 of 5 **short listing**